entry: %result.addr = alloca ptr, align 8 %public.addr = alloca ptr, align 8 %private.addr = alloca ptr, align 8 %size.addr = alloca i32, align 4 %i = alloca i32, align 4 store ptr %result, ptr %result.addr, align 8 store ptr %public, ptr %public.addr, align 8 store ptr %private, ptr %private.addr, align 8 store i32 %size, ptr %size.addr, align 4 store i32 0, ptr %i, align 4 br label %for.cond for.cond: %0 = load i32, ptr %i, align 4 %1 = load i32, ptr %size.addr, align 4 %cmp = icmp slt i32 %0, %1 br i1 %cmp, label %for.body, label %for.end for.body: %2 = load ptr, ptr %public.addr, align 8 %3 = load i32, ptr %i, align 4 %idxprom = sext i32 %3 to i64%arrayidx = getelementptr inbounds i8, ptr %2, i64 %idxprom %4 = load i8, ptr %arrayidx, align 1 %conv = sext i8 %4 to i32%5 = load ptr, ptr %private.addr, align 8 %6 = load i32, ptr %i, align 4 %idxprom1 = sext i32 %6 to i64for.end: %arrayidx2 = getelementptr inbounds i8, ptr %5, i64 %idxprom1 ret void %7 = load i8, ptr %arrayidx2, align 1 %conv3 = sext i8 %7 to i32%mul = mul nsw i32 %conv, %conv3 %conv4 = trunc i32 %mul to i8 %8 = load ptr, ptr %result.addr, align 8 %9 = load i32, ptr %i, align 4 %idxprom5 = sext i32 %9 to i64%arrayidx6 = getelementptr inbounds i8, ptr %8, i64 %idxprom5 store i8 %conv4, ptr %arrayidx6, align 1 br label %for.inc for.inc: %10 = load i32, ptr %i, align 4 %inc = add nsw i32 %10, 1 store i32 %inc, ptr %i, align 4 br label %for.cond, !llvm.loop !6

CFG for 'mix comp' function